Optimización y computación paralela aplicados a mejorar la predicción de un simulador de cauce de ríos

Adriana Gaudiani[1], Emilio Luque [2], Pablo García [3], Marcelo Naiouf [4], Armando De Giusti[4]

[1] Instituto de Ciencias, Univ. Nac. de General Sarmiento, Argentina
[2] Dpto. de Arquitectura de Computadores y Sistemas Operativos, España
[3] Laboratorio de Hidráulica, Instituto Nacional del Agua, Argentina
[4] III-LIDI, Fac. de Informática, Univ. Nac. de La Plata, Argentina

Resumen Este artículo presenta un método de optimización vía simulación que permite mejorar la predicción de un simulador computacional de cauces de ríos aprovechando los valiosos aportes del cómputo paralelo y distribuido. El problema a tratar para lograr este objetivo es la incertidumbre en los parámetros de entrada al modelo, pero buscar parámetros de simulación óptimos es un gran desafío, ya que el tamaño del espacio de búsqueda hace que sea un problema combinatorio con un alto costo de cómputo. La solución propuesta utiliza técnicas de optimización en el análisis de simulaciones, reduciendo el espacio de búsqueda e identificando regiones que permitan encontrar la configuración óptima en tiempos razonables. Se obtiene una muy buena aceleración del procesamiento que es potenciado al implementar el método con técnicas del cómputo de alto rendimiento (HPC).

Keywords: simulación computacional; optimización vía simulación; procesamiento en paralelo; calibración automática

1. Introducción

Los sistemas físicos son sistemas dinámicos que modifican sus variables constantemente. Un ejemplo es la simulación y predicción de inundaciones, provocadas por el desborde del agua que transporta el cauce de ríos, siendo un tema crítico para la población mundial [1]. Sólo las sucesivas calibraciones y validaciones del modelo permiten manejar errores en la entrada, como ser la incertidumbre en los parámetros, logrando que el modelo se ajuste lo mejor posible al comportamiento real del sistema. Normalmente son tareas muy costosas en recursos humanos, tiempo y dinero.

El objetivo principal de este trabajo es optimizar la predicción de un simulador hidrológico, mediante simulación. El método ofrece una calibración y sintonización del modelo con uso exclusivo de métodos computacionales que fueron posibles al utilizar los recursos que provee el HPC. Los resultados de este trabajo aprovechan y realimentan las investigaciones en curso desarrolladas por

el grupo *High Performance Computing for Efficient Applications & Simulation de la Universidad Autónoma de Barcelona*[1] [2], en colaboración con ingenieros del Instituto Nacional del Agua de Argentina (INA), quiénes desarrollaron y nos facilitaron el simulador EZEIZA para avanzar con esta investigación.

Encontrar la solución óptima de esta manera implica encontrar un vector de parámetros de decisión optimizados que al ser ingresados al simulador, junto con los datos completos de un escenario de simulación, provea la menor diferencia entre los datos simulados y los datos observados. La gran dimensión de estos vectores impide implementar un método de búsqueda exhaustiva. La explosión combinatoria resultante convierten esta búsqueda en un método intratable computacionalmente.

El método que se presenta se sitúa entre los métodos de búsquedas exhaustivas y las heurísticas propias de la optimización, utiliza optimización vía simulación (OvS) y consta de dos etapas. La primera etapa es un proceso iterativo que aprovecha la estructura de vecindad del problema y mediante un método Monte Carlo (MMC) identifica una colección de regiones que acercan la búsqueda a la configuración óptima. Un método de agrupamiento K-Means determina la región prometedora donde buscar la solución. La segunda etapa implementa una búsqueda exhaustiva que encuentra la solución, pero ahora sobre una región mucho más pequeña. A pesar de la reducción del espacio de búsqueda, ambas etapas implementan métodos costosos computacionalmente, que necesitan los recursos del cómputo paralelo y distribuido.

La mejora en la predicción del simulador, con los distintos casos de experimentación que se implementaron, fue del $13 - 19\,\%$ sobre los errores de salida en un tramo del río. La aceleración en el tiempo de ejecución se compone de una ganancia propia del método de reducción del espacio de búsqueda y una ganancia en la implementación con HPC. El método computacional es altamente paralelizable. La aceleración lograda con el método de OvS es de 10X/20X en la experimentación presentada.

El resto del artículo está organizado de la siguiente manera: en la sección 2 se detalla el problema a solucionar. En sección 3 se brindan los detalles del método de optimización vía simulación aplicado. En sección 4 se describe la experimentación y los resultados de aplicar procesamiento en paralelo, finalmente en sección 5 se discuten las conclusiones y trabajos futuros.

2. Identificación del Problema

La simulación del cauce de un río representa eventos de inundación provocado por el desborde del cauce natural de ríos. Estos eventos se describen con modelos matemáticos que se basa en las leyes físicas que gobiernan su comportamiento. La certeza en la salida de estos modelo, independientemente de su dimensión, depende en gran medida en la disponibilidad de datos de entrada actualizados. Por otro lado, los errores en datos y parámetros constituyen la principal fuente

[1] http://grupsderecerca.uab.cat/hpc4eas/

de incertidumbre en los resultados de la simulación [3]. En este trabajo se busca una solución a este problema y provee un método computacional de calibración del modelo, que disminuye la incertidumbre de sus parámetros encontrando un conjunto optimizado de sus valores.

2.1. La Incertidumbre de los Datos

Los modelos hidrológicos necesitan al menos los siguientes datos para computar el nivel del agua y la velocidad del flujo en puntos seleccionados del cauce:

- Las secciones que dividen el dominio del río y los intervalos de discretización.
- Los valores de parámetros que definen el sistema.
- Las condiciones de contorno.
- Las condiciones de borde.

Los parámetros pueden evaluarse a partir de datos de campo, por ejemplo, observaciones hidro-meteorológicas, mapas de la topografía, tipos de suelo, etc. Sin embargo, rara vez están disponibles datos de campo completos para validar y determinar con certeza los parámetros del modelo.

2.2. EZEIZA: El Simulador del cauce del Río Paraná

Se seleccionó el simulador de cauces Ezeiza, el cual fue de sumo interés por ser parte del sistema computacional utilizado para el pronóstico que actualmente lleva adelante el *Servicio de Información y Alerta Hidrológico de la Cuenca del Plata*, correspondiente al INA, brindando alertas a la población que reside en zonas de influencia del Río Paraná. La red de cálculo, representada en la Fig. 1, se divide en 76 secciones transversales, que discretizan el Río Paraná, desde el embalse de Yacyretá (Corrientes) hasta la ciudad de Villa Constitución (Santa Fe). Una amplia información puede encontrarse en [4].

Los parámetros seleccionados como los más sensibles del modelo son los siguientes: *la resistencia hidráulica*, que se representa con el valor del coeficiente de rugosidad de Manning, tanto para el cauce del río como para la planicie adyacente [5], y *la altura de los albardones*.

3. Optimización vía Simulación para mejorar la predicción

La búsqueda de un mejor rendimiento del modelo de simulación consiste en repetidamente analizar su salida y mediante sucesivos cambios en los valores de sus parámetros, continuar hasta obtener la mejor salida simulada. La Optimización vía Simulación (OvS) es un campo emergente que integra técnicas de optimización y de análisis de la simulación, el cual se ha convertido en una poderosa técnica para analizar y optimizar estos modelos. En principio, hay una separación entre el modelo que representa el sistema y el procedimiento que es usado para resolver el problema de optimización definido con este modelo. El modelo puede tener modificaciones sin afectar el proceso de optimización que sigue siendo el mismo [6] [7].

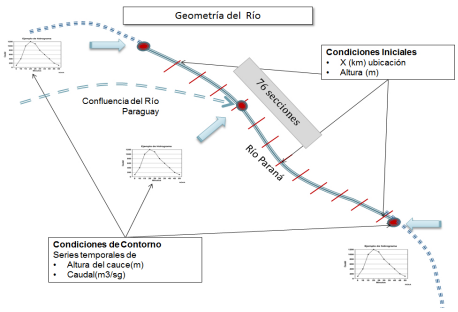

Figura 1. Red de cálculo del modelo hidrodinámico.

3.1. El problema de optimización

El problema de optimización es de múltiples objetivos, siendo necesario optimizar todos los objetivos de forma simultánea, aunque la mayoría de las veces, se encuentran en conflicto entre ellos y se debe encontrar un balance en la solución. Formalmente, la optimización está asociada con la especificación de una función matemática f llamada función objetivo y sus restricciones, de esta forma:

$$\begin{array}{c} max \ / \ min \ f(x) \\ sujeto \ a \ x \ \in S \end{array} \tag{1}$$

donde x es la variable, f es la función ($f : S \rightarrow \Re$), S es el conjunto de restricciones y se cumple que $\exists \ x_0 \ \in S$ tal que $f(x_0 \ \preceq \ f(x) \ \forall \ x \in S$, para minimización, y $f(x_0 \ \succeq f(x) \ \forall \ x \in S$ para maximización.

En el contexto de este trabajo, el problema principal es un *problema de búsqueda*, el cual es considerado como sinónimo de *problema de optimización* y puede expresarse así:

$$encontrar \ x^* \ que \ optimiza \ f(x) \ , \forall \ x \in S$$

donde $f : \Re^p \rightarrow \Re$ es la expresión de la función objetivo o índice de calidad. x^* es el vector de solución óptima. El *espacio de búsqueda* está representado por el dominio $S \subseteq \Re^p$. La especificación de este espacio proviene de un conjunto de restricciones impuestas sobre S, las que restringiendo su tamaño, delimitan la *región de factibilidad*.

El objetivo de la optimización es identificar los valores óptimos de los parámetros que minimizan la diferencia entre la salida del simulador y los datos reales que se miden en el río. Se aplica optimización vía simulación debido a la complejidad del problema de optimización cuya función objetivo no puede resolverse analíticamente.

Ezeiza es considerado como una caja negra en el proceso de simulación, el cual requiere ser alimentado con cada escenario de simulación y el modelo devuelve los datos simulados (de salida) para el tiempo correspondiente a la simulación.

Con cada ejecución se obtiene un *índice de calidad* de la simulación, el cual mide las diferencias entre los valores simulados y los observados.

En cada una de las 76 secciones que dividen el dominio del Río Paraná se determinan los valores de los parámetros del modelo utilizados en la calibración: *Manning de cauce, Mc, y Manning de planicie, Mp, y altura de los albardones, Al*. Sus valores son reales y están definidos sobre intervalos continuos pudiendo tomar infinitos valores dentro de dichos intervalos. Los rangos asociados a cada parámetro deben ser discretizados. Tratar estos intervalos continuos como discretos nos permite ver el problema de optimización como un problema discreto combinatorio.

En la tabla 1 se muestra el excesivo crecimiento de la cantidad de escenarios en función de las secciones que se consideran, considerando solamente Mc y Mp. La explosión combinatoria y el enorme espacio de búsqueda caracteriza el problema. La cantidad de escenarios en función del número de secciones considerada (NSC), se obtiene como $(Mp * Mc)^{NSC}$.

Tabla 1. Explosión combinatoria de escenarios en función de las secciones consideradas

Cardinalidad		**Número de Secciones Consideradas**					
Mc	Mp	2	4	6	8	10	76
2	4	64	4096	262144	1,7 E+07	1,1 E+09	4,3 E+68
3	5	225	50625	1,1 E-07	2,6 E+09	5,8 E +11	2,4 E+89
4	5	400	160000	6,4 E+11	2,6 E+10	1,0 E+13	7,6 E+98
5	6	625	390625	2,4 E+08	1,5 E+11	9,5 E+13	1,8 E+106

3.2. Métricas de Evaluación de la Optimización via Simulación de Ezeiza

El índice utilizado como indicador del rendimiento general de la simulación, IC, depende de los datos de salida simulados, Q_{sim}, y los datos observados, Q_{obs}, en cada una de las 15 estaciones de monitoreo ubicadas sobre el dominio del río. El período de simulación, n, es medido en días. La siguiente ecuación define el índice de calidad de cada escenario de simulación, en función de los índices medidos en cada estación Ie_j:

$$IC = \frac{\sum_{j=1}^{15} Ie_j}{15} \tag{2}$$

$$donde\ Ie_j = \sqrt{1/n \cdot \sum_{i=1}^{n} (Q_{sim}^i - Q_{obs}^i)_j^2} \tag{3}$$

Un aspecto interesante de esta mejora es brindar un escenario de simulación que mejore los resultados que el INA logra actualmente con su escenario de

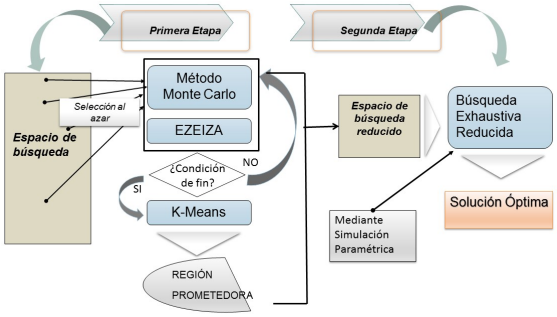

Figura 2. Método de Optimización vía Simulación de Dos Etapas.

pronóstico. Es necesario entonces definir un índice que mida esta mejora, IM, siendo su expresión:

$$IM = (\sharp\ Estaciones\ :\ (IC_j^{Sim} < IC_j^{INA})_{j=1}^{15} \tag{4}$$

IM mide la cantidad de estaciones cuyo índice de calidad, IC_j, sea menor que el logrado con el escenario INA, IC_{INA}, para las 15 estaciones de monitoreo.

3.3. Esquema de reducción propuesto

El método completo de OvS para sintonizar Ezeiza se grafica en la Fig. 2.

En una primera etapa, una combinación del método Monte Carlo + K-means, devuelve un espacio de búsqueda acotado que contiene los datos de los escenarios de simulación con mejores valores promedio del índice de calidad, índice que mide la calidad de simulación con cada escenario. La técnica implementada es un MMC basado en el algoritmo Metrópolis [8] combinado con un método K-means. El MMC selecciona, en cada iteración, 100 escenarios al azar del espacio total de posibles escenarios de simulación que pueden ejecutarse con Ezeiza. Selecciona los que tienen la mejor media y desviación standard que los ya acumulados. Luego el método K-Means identifica conjuntos de regiones prometedoras delimitadas por hiperplanos establecidos por los escenarios encontrados. De esta manera, se obtiene un nuevo espacio de búsqueda reducido. Sobre esta zona se implementa la segunda etapa del método que consiste en un método de búsqueda exhaustiva encontrando así la solución óptima en la región de factibilidad final. Se ofrece más información en [9].

4. Experimentación

A fin de validar la metodología propuesta, se presentan los resultados de una de las experimentaciones realizadas, basadas en simulaciones de $n = 365$ días,

correspondientes al año 1999, período caracterizado por bajantes y subidas en los niveles del cauce del río Paraná.

Los rangos del dominio de los parámetros son:
- Valores de Mc en [0,015 . . . 0,045], tomado en pasos de 0,015. #Mc = 3
- Valores de Mp en [0,2 . . . 0,4], tomado en pasos de 0,1. #Mp = 4
- Valores de Al en [0 . . . 15], tomado en pasos de 5. #Al = 4

Se utilizaron 3 secciones ubicadas en el bajo Paraná: 72, 74 y 76. El espacio de búsqueda, bajo estas condiciones, tiene $(4 \star 3 \star 4)^3$, o sea 110.592 configuraciones diferentes.

La expresión del índice de calidad IC es la siguiente:

$$Minimizar\ IC\ :\ f(Q_{sim},\ Q_{obs}) \tag{5}$$

$$\begin{aligned} donde \quad & 0{,}2 \leq Mp \leq 0{,}4 \\ & 0{,}015 \leq Mc \leq 0{,}045 \\ & 0\ \leq Al \leq\ 15 \end{aligned} \tag{6}$$

La expresión del índice IM es la siguiente:

$$\begin{gathered} Maximizar\ IM\ :\ f(Q_{sim},\ Q_{obs},\ Q_{INA}) \\ tal\ que\ \tfrac{1}{k} \cdot \textstyle\sum (IC_{\widehat{e}}^{Sim} - IC_{\widehat{e}}^{INA}) < tol \end{gathered} \tag{7}$$

donde k representa la cantidad de estaciones $\widehat{e}$ para las cuales se cumple:

$$IC_{\widehat{e}}^{Sim} - IC_{\widehat{e}}^{INA}\ <\ tol$$

donde *tol* es una tolerancia que se estableció en 0.1, según se acordó con los especialistas del INA. Se mantienen las restricciones (6) sobre las variables de decisión.

En la primera etapa, el MMC procesa 52 conjuntos de 100 escenarios para devolver el mejor conjunto de 100 puntos, o sea el que contiene el mínimo valor de IC. Se necesitaron 5200 ejecuciones del simulador para obtenerlo. Luego el método K-Means identifica 3 grupos, que se muestran en la Fig. 3. El grupo identificado con las mejores configuraciones contiene 8 puntos con valores mínimos. Estos puntos son 8 vectores de variables de decisión, que determinan los rangos de valores acotados que éstas pueden tomar y definen así el nuevo espacio de búsqueda reducido, sobre el cual se realiza la búsqueda exahustiva. El nuevo espacio tiene 192 combinaciones.

La Fig. 4 compara los IC obtenidos para cada estación de monitoreo, con el escenario óptimo obtenido con el método de OvS y el escenario INA. En este gráfico puede apreciarse las mejoras logradas en tres ciudades del tramo inferior del Río Paraná.

La solución resultante es la configuración óptima mostrada en la tabla 2. Esta solución corresponde al escenario que tiene el *mínimo IC del conjunto y a la vez maximiza IM*. El índice IC óptimo del método es 0.801 y el valor de IM es: 3 estaciones. La configuración obtenida se muestra en la tabla 3.

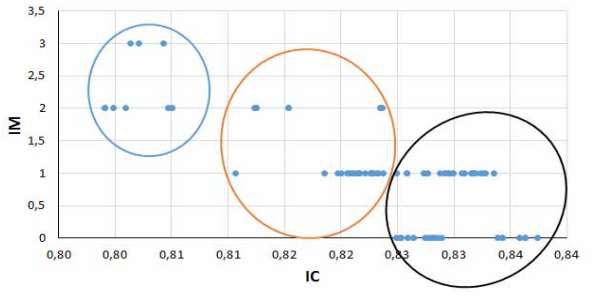

Figura 3. Grupos de configuraciones identificados por K-Means, considerando $IC\&IM$.

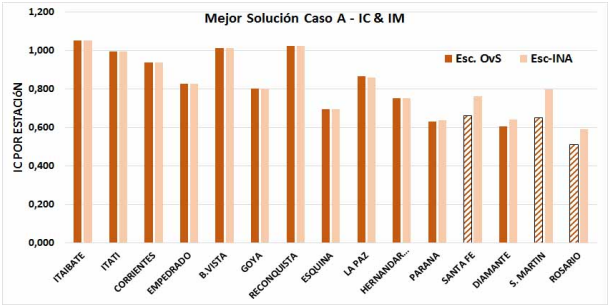

Figura 4. Comparación entre los valores de IC para el Esc.OvS en cada estación y la configuración INA tomando como objetivos $IC\&IM$.

4.1. Implementación en Paralelo del Método OvS

Todas las simulaciones del método de OvS que se describen en este capítulo fueron implementados y ejecutados en equipamiento del Instituto de Investigación en Informática LIDI (III-LIDI) de la UNLP. Se utilizó un cluster de multicores HP Blade que consiste de 16 nodos, donde cada uno cuenta con 2 procesadores quad-core Intel Xeon e5405 de 2.0GHz. A su vez, cada núcleo cuenta con una caché L1 privada de 64KB (dividida en 32KB para instrucciones y otros 32KB para datos) y una caché L2 de 6MB compartida de a pares de núcleos. Con res-

Tabla 2. Solución óptima del método con objetivos:$IC + IM$.

Solución Óptima - $IC + IM$			
Sección	**Mc**	**Mp**	**Al**
1	0,02	0,2	5
2	0,02	0,2	5
3	0,04	0,2	0

Tabla 3. Mejora lograda en las estaciones del cauce.

Estación	**Escenario**		**Mejora**
	OvS	**INA**	
Santa Fe	0,662	0,764	13 %
San Martín	0,651	0,799	19 %
Rosario	0,512	0,59	13 %

Tabla 4. Tiempo de ejecución y ganancia lograda en cada etapa del método de OvS.

CASO A	Secuencial		16 cores		Ganancia 1000 cores	
Simulaciones	minutos	días	minutos	días	speedup	minutos
110592	243302	169	15896	11	15,306	240
5200	11440	8	756	0,5	15,132	11
8						
192	422	0,3	27	0	15,63	0,4

pecto a la memoria RAM disponible, 10 nodos cuentan con 10GB mientras que el resto dispone de 2GB. El sistema operativo instalado es GNU/Linux Debian 6.0 de 64 bits (versión del kernel 2.6.32).

4.2. Ganancia lograda con el uso HPC

El método propuesto permitió lograr una significativa ganancia en la eficiencia de la implementación, que se puede analizar en dos niveles de su desarrollo. En un primer lugar, el método de OvS permitió reducir del espacio de búsqueda y, en consecuencia, obtener una aceleración en cantidad de cómputo medida en simulaciones. En segundo lugar, la implementación del método aplicando técnicas de procesamiento en paralelo brindó su propia aceleración, que se suma a la primera.

La ganancia del MMC y la ganancia global están representadas en la Fig. 5.

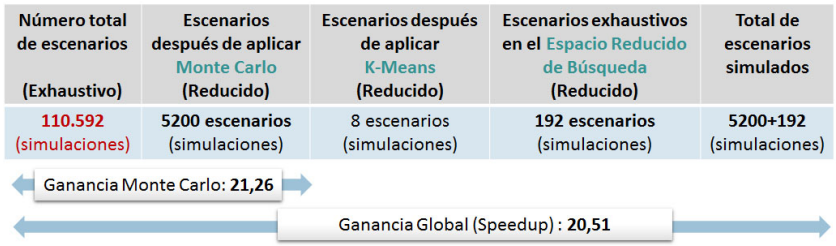

Figura 5. Ganancia lograda con el método implementado

La tabla 4 compara los tiempos de ejecución del método, en el caso de la implementación de todo el espacio de búsqueda y en las dos etapas del método de OvS.

5. Conclusiones y líneas futuras

El método propuesto en este trabajo permite realizar una calibración automática de un simulador de cauces de ríos, brindando una técnica que mediante el uso exclusivo de cómputo y procesamiento paralelo y distribuido, mejora la

predicción del simulador. Las mejoras en la predicción del nivel del cauce en la ciudades del bajo Paraná entre 13-19 %. Estos valores se traducen en diferencias de altura entre 40 a 60 cm que pueden ser la diferencia entre la pérdida de bienes y de ganado, o el aviso temprano para evitarlo. La implementación en paralelo del método es fácilmente escalable, o sea cuando más recursos de cómputo se tengan más ganancia se obtiene, como se ve en la tabla 4 al utilizar 1000 cores. Pero el propósito de este trabajo es aportar una mejora a la predicción del simulador con la menor cantidad de recursos computacionales posible, objetivo que se alcanza con los dos niveles de speedup logrados. Una línea de trabajo futuro está centrada en la mejora del método de OvS para ajustar la predicción en los momentos que ocurre una crecida rápida del cauce del río, como también brindar períodos de validez del conjunto de parámetros óptimos encontrados por el método. El objetivo sería recalibrar automáticamente y predecir por períodos más extensos sin perder la calidad lograda en la simulación.

Agradecimientos

Esta investigación ha sido financiada por el MINECO España, bajo contrato TIN2014-53172-P y fue apoyado por el programa de investigación del Instituto de Investigación en Informática III-LIDI, Facultad de Informática, UNLP.

Referencias

1. S. Jonkman and J. Vrijling, "Loss of life due to floods," *Journal of Flood Risk Management*, vol. 1, no. 1, pp. 43–56, 2008.
2. E. Cabrera, E. Luque, M. Taboada, F. Epelde, and M. Iglesias, "Optimization of emergency departments by agent-based modeling and simulation," in *Information Reuse and Integration (IRI), 2012 IEEE 13th International Conference on*, 2012, pp. 423–430.
3. J. Warmink, J. Janssen, M. Booij, and M. Krol, "Identification and classification of uncertainties in the application of environmental models," *Environmental Modelling & Software*, vol. 25, no. 12, pp. 1518 – 1527, 2010.
4. A. Menéndez, "Three decades of development and application of numerical simulation tools at ina hydraulics lab." *Mecánica Computacional*, vol. XXI, pp. 2247–2266, 2002.
5. G. J. Arcement and V. R. Schneider, *Guide for selecting Manning's roughness coefficients for natural channels and flood plains*, ser. USGS Water-Supply Paper 2339. US Government Printing Office Washington, DC, USA, 1989.
6. L.-F. Wang and L.-Y. Shi, "Simulation optimization: A review on theory and applications," *Acta Automatica Sinica*, vol. 39, no. 11, pp. 1957 – 1968, 2013.
7. J. F. Santucci and L. Capocchi, "Optimization via simulation of catchment basin management using a discrete-event approach," *Simulation*, vol. 91, no. 1, pp. 43–58, Jan. 2015, iSSN:0037-5497.
8. D. P. Kroese, T. Taimre, and Z. I. Botev, *Handbook of Monte Carlo Methods*. John Wiley & Sons, 2013, vol. 706.
9. A. Gaudiani, E. Luque, P. García, M. Re, M. Naiouf, and A. De Giusti, "Computing, a powerful tool for improving the parameters simulation quality in flood prediction," *Procedia Computer Science*, vol. 29, no. 0, pp. 299 – 309, 2014.